

## AMENDMENTS TO THE CLAIMS

1. (Previously Presented) A method for allowing multiple applications to cooperatively access the same hardware resource, said method comprising:
  - a) registering a callback instruction for a first application that is using said hardware resource;
  - b) invoking said callback instruction to notify said first application of a request from a second application for the same said hardware resource; and
  - c) yielding said hardware resource to said second application provided said first application grants said request.
2. (Previously Presented) The method as recited in Claim 1 wherein said hardware resource comprises interface circuitry coupled to multiple ports.
3. (Previously Presented) The method as recited in Claim 1 further comprising:
  - registering said first application as a passive application, wherein a passive application defines said callback instruction.
4. (Original) The method as recited in Claim 1 wherein said step b) is performed responsive to said request from said second application.

5. (Previously Presented) The method as recited in Claim 1 further comprising:

providing notice to said first application that said second application is finished using said hardware resource, said notice indicating said hardware resource is available.

6. (Previously Presented) The method as recited in Claim 1 wherein said step c) further comprises:

- c1) closing said hardware resource for said first application; and
- c2) conducting procedures for shutting down said first application.

7. (Original) The method as recited in Claim 1 wherein a response granting said request is a Boolean true, and wherein a response denying said request is a Boolean false.

8. (Currently Amended) A method for allowing multiple applications to cooperatively access a same serial port, said method comprising:

- a) opening said serial port for a first application, wherein said opening comprises registering a callback instruction for said first application;
- b) receiving a request for the same said serial port from a second application;
- c) invoking said callback instruction ~~responsive to said request to~~ notify said first application of said request from said second application, wherein said invoking comprises:

c1) sending notice to said first application of said request; and  
c2) receiving from said first application a response to said notice; and  
d) yielding the same said serial port to said second application provided said response from said first application grants said request and otherwise maintaining said serial port for said first application.

9. (Previously Presented) The method as recited in Claim 8 further comprising:

registering said first application as a passive application.

10. (Previously Presented) The method as recited in Claim 8 wherein said step d) further comprises:

receiving from said first application a response denying said request.

11. (Previously Presented) The method as recited in Claim 8 wherein said step d) further comprises:

returning an error message to said second application when said serial port is not yielded to said second application.

12. (Previously Presented) The method as recited in Claim 8 further comprising:

providing notice to said first application that said second application is finished using said serial port, said notice indicating said serial port is available.

13. (Previously Presented) The method as recited in Claim 8 wherein said step c) further comprises:

- c3) closing said serial port for said first application; and
- c4) conducting procedures for shutting down said first application.

14. (Original) The method as recited in Claim 8 wherein a response granting said request is a Boolean true, and wherein a response denying said request is a Boolean false.

15. (Currently Amended) A portable computer system comprising:

- a bus;
- a serial port coupled to said bus;
- a processor coupled to said bus; and
- a memory coupled to said bus, said memory comprising instructions for implementing a method for allowing multiple applications residing on said computer system to cooperatively access the same said serial port, said method comprising:

- a) opening said serial port for a first application, wherein said opening comprises registering a callback instruction for said first application;

- b) receiving a request for the same said serial port from a second application;

- c) invoking said callback instruction ~~responsive to said request to~~  
notify said first application of said request from said second application,  
wherein said invoking comprises:

c1) sending notice to said first application of said request; and  
c2) receiving from said first application a response to said notice; and  
d) yielding said serial port to said second application provided said response from said first application grants said request and otherwise maintaining said serial port for said first application.

16. (Previously Presented) The computer system of Claim 15 wherein said method further comprises:

registering said first application as a passive application.

17. (Previously Presented) The computer system of Claim 15 wherein said step d) of said method further comprises:

receiving from said first application a response denying said request.

18. (Previously Presented) The computer system of Claim 15 wherein said step d) of said method further comprises:

returning an error message to said second application when said serial port is not yielded to said second application.

19. (Previously Presented) The computer system of Claim 15 wherein said method further comprises:

providing notice to said first application that said second application is finished using said serial port, said notice indicating said serial port is available.

20. (Previously Presented) The computer system of Claim 15 wherein said step c) of said method further comprises:

- c3) closing said serial port for said first application; and
- c4) conducting procedures for shutting down said first application.

21. (Original) The computer system of Claim 15 wherein a response granting said request is a Boolean true, and wherein a response denying said request is a Boolean false.